Aspectos Técnicos

Toda la información descrita en el siguiente documento es de carácter obligatorio incluirlo en la tarea, por favor incluirlos.

Main

Structs definidos en main.h

```
typedef struct {
    Tablero* tablero; // Puntero al struct Tablero
    Ingrediente** inventario;
    Pedido* pedidos;
    int turnos_restantes;
    int dificultad;
} Juego;
typedef struct {
    int x;          // Posicion X en el tablero
    int y;          // Posicion Y en el tablero
    int en_llamas;  // 1 si el jugador esta en llamas (por
       atravesar estacion incendiada)
} Jugador;
```

Tablero

- Estructura: Puntero triple: Representa una matriz 2D dinámica donde cada celda almacena un puntero a un elemento (estación, jugador o espacio vacío). **Su uso es obligatorio**

- Struct definido en tablero.h

```
typedef struct {
    void*** celdas;    // Matriz 2D de elementos (puntero triple
       )
    int filas;
    int columnas;
} Tablero;
```

- Funciones en `Tablero.c/h`:

```
void inicializarTablero(Tablero* tablero, int filas, int
   columnas)
void mostrarTablero(Tablero* tablero);
void actualizarCelda(Tablero* tablero, int x, int y, void*
   elemento)
```

Inventario

- Estructura: Puntero doble `void**`, que apunta a una lista estática de ingredientes y, eventualmente, el extintor.

- Funciones en `inventario.c/h`:

```
void crearInventario();
void agregarIngrediente();
void eliminarIngrediente();
void verInventario();
```

- Struct definido en `inventario.h`:

```
typedef struct {
    char nombre[30];     // Ej: "pan", "pollo"
    int estado;    // Ej: "0 = crudo", "1 = cortado", etc.
    int es_extintor;    // 1 si es extintor, 0 si es ingrediente.
    int turnos_elaboracion; // Turnos que tarda en prepararse.
    int prob_incendio;      // Probabilidad de causar incendio.
} Ingrediente;
```

Estaciones y Acciones

- Estructura: Cada estación tendrá asociada una función a través de un puntero a función. Al interactuar con una estación, se invoca automáticamente la función correspondiente.

- Structs definidos en **acciones.h**

```
typedef struct {
    char simbolo;               //'T' (Tabla), 'C' (Cocina), 'A' (Almacen), 'E' (Extintor)
    void (*accion)(void*, int, int); // Puntero a funcion (ej: cocinar, cortar)
    int en_llamas;              // 1 si hay incendio, 0 si no
    int turnos_inhabilitada;    // Para estaciones apagadas con extintor
} Estacion;
```

```
typedef struct {
    char nombre_plato[50]                    //Ej: McCharly
    Ingrediente** ingredientes_requeridos; // Lista de ingredientes necesarios
    int completado;                          // 1 si entregado, 0 si pendiente
} Pedido;
```

- **Funciones clave (acciones.h)** vinculadas a estaciones

```
void cortar(void* contexto, int x, int y);    // Tabla de cortar
void cocinar(void* contexto, int x, int y);   // Cocina
void buscar_ingrediente(void* contexto, int x, int y); //Almacen
void apagar_incendio(void* contexto, int x, int y); // Extintor
void entregar_pedido(void* contexto);
```